поиск в СС всегда есть какое-то сопоставление), но не технически (в том смысле, что любой требуемый поиск может быть выражен операцией сопоставления в некотором универсальном языке представления знаний). Именно поэтому базовыми для СС здесь названы не операции сопоставления с образцом, а гораздо более примитивные операции перехода по сети. Фактически совокупность последних в каждом конкретном случае поиска реализует то или иное требуемое сопоставление.

4.2.5. Исторические примеры семантических сетей

Исследования по СС начались с работы М. Р. Квиллиана в 1966 г.[1], в которой он попытался формально представить семантику английских слов. М. Р. Квиллиан ввел термин «семантическая память» для системы, в основе которой лежал формальный язык, выражениями которого были графы с помеченными вершинами и дугами. Точнее формальные выражения этого языка были составлены из вершин, которым отвечали «понятия-слова», и из «ассоциативных дуг», соединяющих вершины. Эти дуги фактически исполняли роль указателей, с помощью которых одни слова-понятия ссылаются на другие. Эта структура взаимных ссылок между словами-понятиями и дает определение исходного слова-понятия. Аналогичное наблюдается в толковых словарях, в которых толкование данного слова конструируется из последовательности слов, толкования которых разбросаны в разных частях словаря.

Одним из главных намерений Квиллиана было развитие дедуктивных возможностей модели семантической памяти на основе свойств отношений *класс — подкласс* и *модификации. Понятие* определялось в терминах более общего понятия (т.е. первое есть подкласс второго) и с помощью модифицирующего свойства, являющегося комбинацией *атрибут — значение атрибута*. При этом свойство, истинное для элементов класса, наследовалось элементами любого его подкласса. Таким образом, СС Квиллиана представляла собой комбинацию двух механизмов: таксономической иерархии, основанной на отношении *класс — подкласс*, и описания свойств (как пара «атрибут — значение») элементов класса.

СС этого типа имела ограничения, частично вызываемые тем, что набор типов связок (дуг) был недостаточен и поиск не принимал в расчет их смысл (т.е. не было интерпретирующих программ для связок). Однако идеи Квиллиана сыграли важную роль для дальнейшего развития СС.

Важное значение для развития теории СС имели работы Г. Хендрикса[2].

Как было сказано выше, вершинами СС могут быть понятия (объекты), события (действия), свойства (места), а дугами — отношения, которые делятся на лингвистические, логические, теоретико-множественные, квантификационные. В свою очередь лингвистические отношения делятся на: глагольные (время, вид, род, число, залог, наклонение); атрибутивные (цвет, размер, форма, модификация); логические (дизъюнкция, конъюнкция, отрицание, импликация); теоретико-множественные. Квантификаци-

[1] *Геловани В. А. и др.* Указ. соч.

[2] Там же.

онные делятся на логические кванторы общности и существования, нелогические кванторы (много, несколько) и числовые характеристики.

В СС, предложенную Г. Хендриксом, можно записывать таксономическую информацию о процессах и процедурах, модальности (мнения, желания), информацию о границах локальных контекстов и т.п. Структура этой сети обладает большой выразительностью, в ней наиболее полно нашли отражение основные принципы построения таких сетей, и, кроме того, она стали базой для дальнейшего развития фреймовой модели представления знаний в виде сети.

Рассмотрим пример сети Г. Хендрикса, на котором будут пояснены основные понятия структуры СС (рис. 4.27).

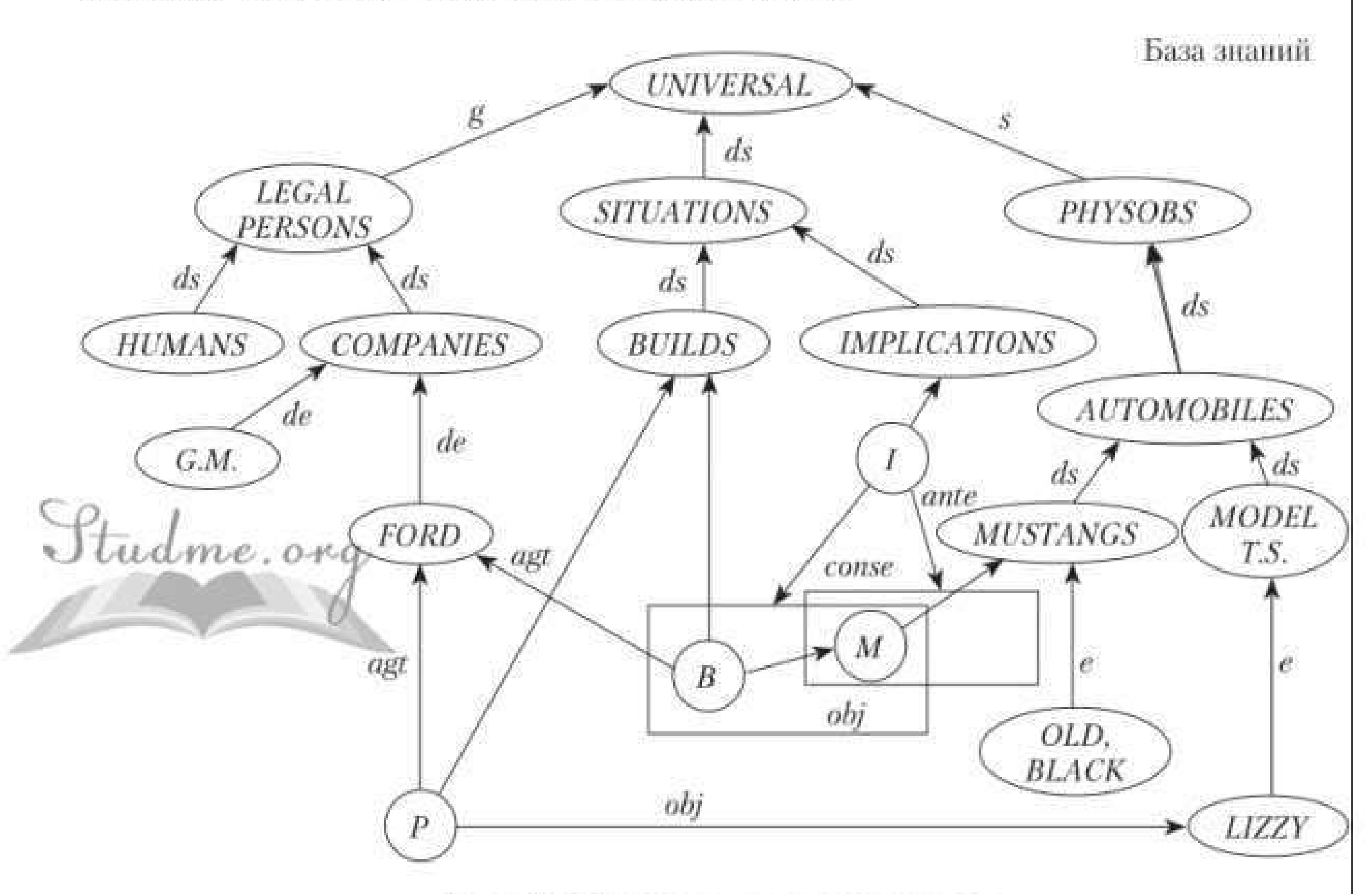


Рис. 4.27. **Пример сети Хендрикса**

На этом рисунке вершина *UNIVERSAL* обозначает универсальный класс (универсум), который объединяет различные объекты. Класс *UNIVERSAL* содержит три подкласса: *LEGAL PERSONS* (юридические лица), *SITUATIONS* (ситуации) и *PHYSOBS* (физические объекты).

Метка *s* обозначает *отношение включения*, а метка *ds* кроме отношения включения подразумевает, что элементы одного класса отличны от элементов другого (например, элементы подкласса *SITUATIONS* отличны от элементов *LEGAL PERSONS* и *PHYSOBS*). Метка *e* обозначает элемент класса. Метка *de* обозначает, что данный элемент отличен от всех других элементов, обозначения которых указаны другими *e*(*de*)-дугами для данного класса (например, *FORD* и *G. M.* (*General Motors*) обозначают различные элементы класса *COMPANIES* (компании)). Метки *agt* и *obj* применяются для обозначения семантических падежей: *агент*, т.е. лицо, вызывающее действие, и *объект* — предмет, подвергающийся действию.

соответственно. Метки *conse* и *ante* обозначают соответственно консеквент и антецедент импликации.

Важную роль в развитии СС сыграла также идея их представления в виде блоков, когда каждый блок имеет свою вершину входа (например, вершины *P*, *B* и *M* для трех блоков соответственно).

Блок — одна из основных структурных единиц сети (наравне с вершинами и дугами). Каждая вершина и каждая дуга сети принадлежит одному или нескольким блокам (например, вершина *M* принадлежит трем блокам). Дуги, принадлежащие блоку, задаются метками внутри прямоугольника (например, дуга, связывающая *M* с *MUSTANGS*, принадлежит блоку, ограниченному самым маленьким прямоугольником, но не принадлежит блоку, ограниченному большим прямоугольником) (см. рис. 4.27). Таким образом, разбиение сети на блоки дает возможность для образования иерархии блоков.

К двум блокам с вершинами *B* и *M* добавлены классы *IMPLICATIONS* (импликация) и *BUILDS* (строит). Получается утверждение: «Если *M* является элементом класса *MUSTANGS*, то для некоторой ситуации *B* из класса *BUILDS* агентом *B* является *FORD*, а объектом *B* является *M*. На языке логики исчисления предикатов первого порядка это утверждение запишется так:

$$(\forall \mathrm{M} \subset \mathrm{MUSTANGS})(\exists \mathrm{B} \subset \mathrm{BUILDS})\ \mathrm{agt}(\mathrm{B}) = \mathrm{FORD} \wedge \mathrm{obj}(\mathrm{B}) = \mathrm{M}.$$

Таким образом, в сети имеется возможность представлять комбинацию кванторов общности и существования. Иногда удобно скомбинировать несколько блоков в один «пучок» блоков. Такую комбинацию блоков Хендрикс назвал «видом» (*vista*). Эта идея применяется при извлечении информации из сети, когда выгодно иметь только определенный список блоков — «вид», из которых и выбирается информация, т.е. говорят, что вид «видим» в данный момент (рис. 4.28).

Все структуры, лежащие вне этого вида, и, следовательно, заключенная в них информация, будут игнорироваться. Механика разбиения сети на блоки позволяет создавать иерархию видов, причем вид наследует информацию родительского вида. На рис. 4.28 дан пример использования иерархии видов. Жирные стрелки указывают на наследование способности выборки информации. Внизу от каждого блока помещен список блоков, составляющих вид. Таким образом, из вида V_1 «видна» глагольная часть фразы: «Некоторый человек *M* владеет машиной *C*», а сама фраза видна из V_6.

В рассматриваемой СС могут присутствовать блоки логических операций: отрицания, дизъюнкции и импликации. Эти блоки особенно важны для логической дедукции при построении систем принятия решений. Что касается конъюнкции, то она реализуется в сети с помощью механизма наследования.

В СС отрицание включает описание некоторого множества понятий и отношений, имеющих место на рис. 4.29, дана сеть, представляющая утверждение: «Компания G.M. не строила OLD.BLACK».

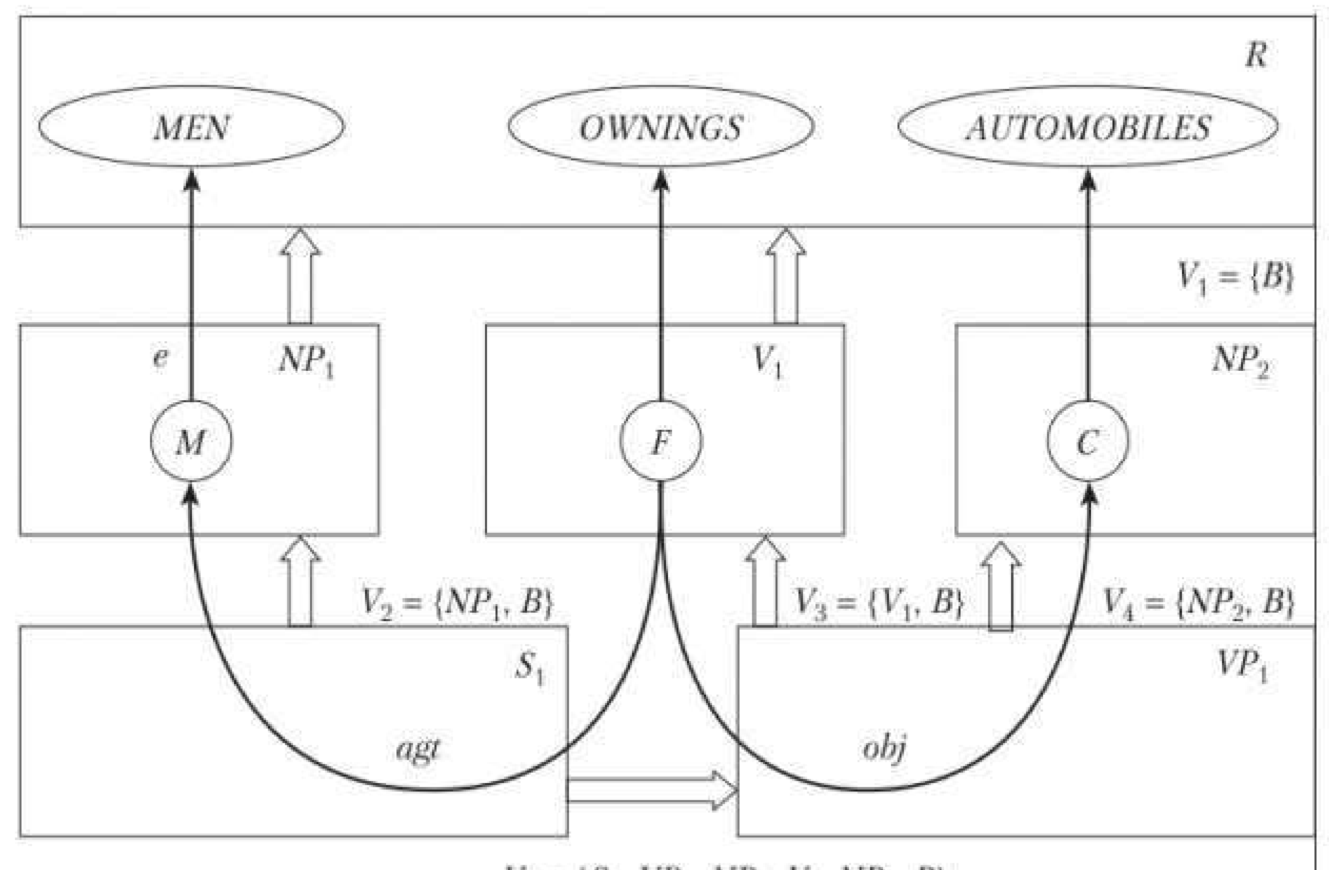


Рис. 4.28. **Пример иерархии блоков в семантической сети**

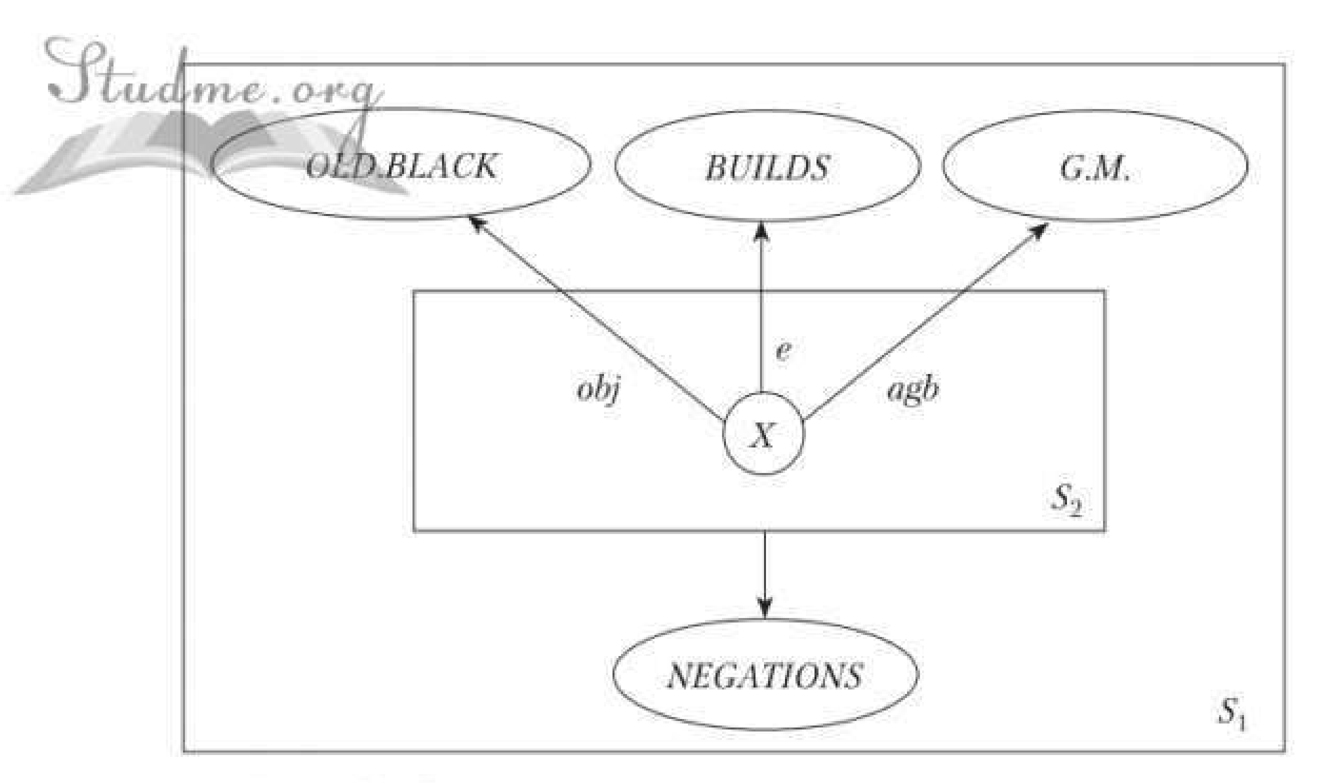


Рис. 4.29. **Семантическая сеть утверждения с отрицанием**

Одной из важных особенностей сети является ее способность представлять произвольно сгруппированные кванторы общности и существования. Квантор существования является «встроенным» понятием в том смысле, что если некоторый элемент (вершина или дуга) входит в данный блок, то соответствующее понятие существует в мире, который данный блок представляет.

Квантор существования и отрицание могут использоваться для представления любой формулы с квантором общности $\forall(x \subset X)(P(x))$, имея в виду следующее преобразование:

$$\forall(x \subset X)P(x) = [\overline{\overline{\forall(x \subset X)P(x)}}] = [\overline{\exists(x \subset X)\overline{P(x)}}].$$

На рис. 4.30 приведен пример представления в СС квантора всеобщности.

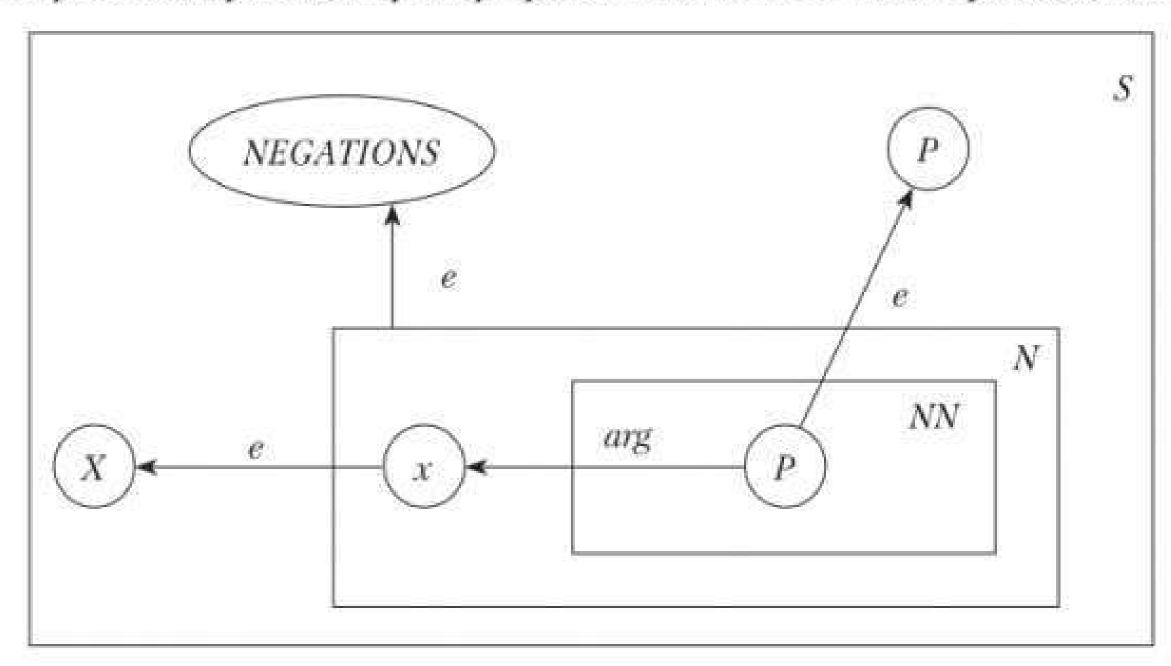


Рис. 4.30. **Семантическая сеть квантора всеобщности**

Дальнейшее развитие идеи разбиения сети на блоки находит свое выражение в использовании многосортной логики предикатов для более эффективного представления знаний. Основная идея состоит в том, что область (универсум) рассуждений *D* распадается на ряд различных именованных подмножеств, называемых семантическими категориями (Дж. Минкер и Дж. Мак-Скимин)[1]. В сущности понятие «семантической категории» эквивалентно понятию «сорта», применяемому в логике. Семантическая категория — это имя, данное совокупности определенных элементов в области рассуждений. Например, «мужчина» — это семантическая категория, состоящая из имен всех мужчин, имеющихся в БД.

Каждый элемент рассуждений *D* может быть задан как список свойств, характеризующих этот элемент. Каждый предикат, заданный на *D*, допускает в качестве аргументов только те элементы, которые обладают определенными свойствами.

В дальнейшем универсум рассуждений будет считаться универсальной семантической категорией, обозначаемой UNIV.

Имена, даваемые категориями, могут определяться произвольно и являются синтаксическими объектами. Для задания теоретико-множественных отношений между семантическими категориями строится граф семантических категорий или просто семантический граф.

Семантический граф *G* есть конечный граф, вершинами которого служат семантические категории. Дуги *G* указывают теоретико-множествен-

---

[1] *Геловани В. А. и др.* Указ. соч.

ные отношения: вложения, равенства, пересечения, объединения, дополнения. Висящие вершины графа G называются примитивными категориями.

Из категорий с использованием операций пересечения, объединения и дополнения можно получить так называемые булевы категориальные выражения, которые сильно повышают выразительные возможности семантических категорий (Дж. Минкер и Дж. Мак-Скимин) (Геловани, Башлыков, 2001). Примеры булевых категориальных выражений: растение, животное, студент молодой.

Таким образом, в СС можно выделить следующие структурные компоненты:

- семантический граф, который определяет теоретико-множественные отношения между семантическими категориями;
- базу данных, состоящую из утверждений и общих правил (законов), представленных в виде дизъюнктов;
- пространство семантических форм, которое определяет семантические ограничения, наложенные на аргументы реляционных n-ок;
- словарь, который определяет для каждого элемента области, каждого функционального имени и предикатного имени ту семантическую категорию, членом которой он является.

В настоящее время аппарат семантических сетей широко используется в практике представления знаний. Его достоинствами являются: большие выразительные возможности; естественность и наглядность системы знаний, представленной графически; близость структуры сети знаний, семантической структуре фраз естественного языка.

Общая идея использования ориентированных графов для представления знаний по-разному воплотилась у разных исследователей. Этим объясняется отсутствие единой терминологии и формально определяемой семантики для методов представления знаний, основанных на СС. Анализируя примеры, мы видим, насколько субъективными могут быть такие представления о СС. Сколько авторов — столько и вариантов СС.

Такая неоднозначность может рассматриваться и как достоинство, и как недостаток. Достоинство — в большой свободе выражения и открытости СС; недостаток — в сложности их формального анализа и обработки. Поэтому ученые с самого начала пытались ограничить свободу авторов и построить более строгие представления. Рассмотрим некоторые из них.

4.2.6. Методика построения предикатной семантической сети

Выше были рассмотрены несколько типов СС: предикатные, атрибутивные, иерархические и др. Мы рассмотрим пример построения предикатной сети. В качестве примера разберем задачу об обезьяне и бананах.

Вначале следует дать словесное описание ПрО. Обозначим через P_i различные ситуации (предложения), возникающие в пространстве состояний:

Р1. Обезьяна, Ящик и Бананы находятся в различных точках Комнаты — соответственно a, b, c;

Р2. Обезьяна видит Бананы *И* пытается их достать, но Бананы висят высоко;